

AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A computer system for building a health model of software components, comprising one or more computer processors, system memory, and one or more physical computer-readable media having stored thereon computer-executable instructions which, when executed, perform ~~[[the]]~~ a method ~~comprising: of claim 10.~~

creating an inventory of instrumentation of the software components;

mapping each individual instrumentation in the inventory of instrumentation to a state of operation of the software components before the instrumentation is generated, the state of operation before the instrumentation is generated being from a list comprising running, failed, stopped, paused, non-activated, and activated;

mapping each individual instrumentation in the inventory of instrumentation to a state of operation of the software components after the instrumentation is generated, the state of operation before the instrumentation is generated being from a list comprising running, failed, stopped, paused, non-activated, and activated;

analyzing the inventory to identify instrumentation that result in the same transition from one state of operation of the software components to another state of operation of the software components;

grouping the identified instrumentation that result in the same transition from one state of operation of the software components to another state of operation of the software components by filtering the instrumentation based upon the state of the operation of the software component before instrumentation was generated and the state of operation after the instrumentation was generated;

generating the health model using the states of operation and at least one transition representing a group of instrumentation from one state of operation of the software components to another state of operation of the software components; and

representing the health model in a table comprising fields including an event group, an event identification, the state before an event occurs, the state after an event occurs, an anti-alert, a remedy, a blame component, and a description.

2. (Original) The system of claim 1 further comprising a database for storing the information about the instrumentation of the software components.

3. (Original) The system of claim 1 further comprising a database for storing the health model.

4. (Original) The system of claim 1 wherein the health model comprises a state diagram with a transition from one state to another state for a group of instrumentation.

5. (Original) The system of claim 1 wherein the instrumentation collector comprises a spreadsheet for manual entry of information about instrumentation of software components.

6. (Previously Presented) The system of claim 1 wherein the creating an inventory comprises a parser automatically parsing a software component to extract information about instrumentation of the software component.

7. (Previously Presented) The system of claim 3 wherein the database also stores the instrumentation information used to generate the health model.

8. (Previously Presented) The system of claim 1 wherein generating the health model comprises an application that generating a state diagram.

9. (Previously Presented) The system of claim 1 wherein the computer-executable instructions comprise a plurality of modules.

10. (Currently Amended) A method for building a health model of a software component, comprising the steps of:

creating an inventory of instrumentation of the software component;

mapping each individual instrumentation in the inventory of instrumentation to a state of operation of the software component before the instrumentation is generated, the state of operation before the instrumentation is generated being from a list comprising running, failed, stopped, paused, non-activated, and activated;

mapping each individual instrumentation in the inventory of instrumentation to a state of operation of the software component after the instrumentation is generated, the state of operation before the instrumentation is generated being from a list comprising running, failed, stopped, paused, non-activated, and activated;

analyzing the inventory to identify instrumentation that result in the same transition from one state of operation of the software component to another state of operation of the software component;

grouping the identified instrumentation that result in the same transition from one state of operation of the software component to another state of operation of the software component by filtering the instrumentation based upon the state of the operation of the software component before instrumentation was generated and the state of operation after the instrumentation was generated; {{and}}

generating the health model using the states of operation and at least one transition representing a group of instrumentation from one state of operation of the software component to another state of operation of the software component; and

representing the health model in a table comprising fields including an event group, an event identification, the state before an event occurs, the state after an event occurs, an anti-alert, a remedy, a blame component, and a description.

11. (Cancelled)

12. (Previously Presented) The method of claim 10 wherein the step of creating an inventory of instrumentation of the software component comprises parsing the software component to extract information about instrumentation of the software component.

13. (Original) The method of claim 10 further comprising the step of determining states of operation of the software component.

14. (Original) The method of claim 13 wherein the step of determining states of operation of the software component comprises determining a stopped state.

15. (Original) The method of claim 13 wherein the step of determining states of operation of the software component comprises determining a running state.

16. (Original) The method of claim 13 wherein the step of determining states of operation of the software component comprises determining a failed state.

17. (Original) The method of claim 10 further comprising the step of adding instrumentation where there is none to indicate an occurrence of a transition from a failed state of operation to a running state of operation.

18. (Original) The method of claim 10 further comprising the step of adding instrumentation where there is none to indicate an occurrence of a transition from a running state of operation to a failed state of operation.

19. (Original) The method of claim 10 further comprising the step of persistently storing the inventory of instrumentation.

20. (Original) The method of claim 10 further comprising the step of persistently storing the generated health model.

21. (Original) The method of claim 10 further comprising revising the instrumentation of the software component.

22. (Original) The method of claim 21 further comprising updating the health model using the revised instrumentation.

23. (Original) The method of claim 22 further comprising generating a new health model.

24. (Original) The method of claim 10 wherein the step of receiving an inventory of instrumentation comprises receiving an inventory of one or more events.

25. (Original) The method of claim 10 wherein the step of receiving an inventory of instrumentation comprises receiving an inventory of one or more performance counters.

26. (Original) The method of claim 10 wherein the step of receiving an inventory of instrumentation comprises receiving an inventory of one or more error messages.

27. (Original) The method of claim 10 wherein the step of receiving an inventory of instrumentation comprises parsing the software component to extract information about instrumentation of the software component.

28. (Original) The method of claim 10 wherein the step of receiving an inventory of instrumentation comprises manually entering instrumentation information in a spreadsheet application.

29. (Original) The method of claim 10 wherein the step of analyzing the inventory comprises determining the state of operation before an instrumentation event occurs and the state of operation after the instrumentation event occurs.

30. (Previously Presented) The method of claim 10 further comprising storing the health model and the instrumentation used to generate the health model in a database.

31. (Original) The method of claim 10 wherein the step of analyzing the inventory to group instrumentation that result in the same transition from one state of operation of the software component to another state of operation of the software component comprises labeling each group of instrumentation as a single transition action from one state of operation of the software component to another state of operation of the software component.

32. (Original) The method of claim 10 wherein the step of analyzing the inventory to group instrumentation that result in the same transition from one state of operation of the software component to another state of operation of the software component comprises using an application to analyze the inventory to group instrumentation that result in the same transition from one state of operation of the software component to another state of operation of the software component.

33. (Original) The method of claim 10 wherein the step of analyzing the inventory comprises determining a component to blame for instrumentation indicating a failure of the software component.

34. (Original) The method of claim 10 wherein the step of generating the health model comprises generating a state diagram.

35. (Original) The method of claim 26 wherein the step of generating a state diagram comprises using an application to generate the state diagram.

36. (Previously Presented) A computer storage medium having computer-executable instructions for performing the method of claim 10.

37 – 40. (Cancelled)

41. (Previously Presented) The method of claim 10, wherein the health model is configured to detect cycles of change in states of operation.

42. (Previously Presented) The method of claim 41, wherein at least one of the cycles of change in states of operation comprises a cycle of failure and recovery.

43 – 44: (Cancelled)